Low-Photon Phase Retrieval

David Barmherzig

January 15, 2020

Collaborators

- Ju Sun, University of Minnesota
- Charles Epstein, Leslie Greengard, Alex Barnett, Jeremy Magland, Flatiron Institute
- Stefano Marchesini, Lawrence Berkeley National Laboratory
- Emmanuel Candès, Stanford University

Coherent Diffraction Imaging (CDI)

The Phase Retrieval Problem in CDI

$$\textbf{Given} \quad \left|\widehat{X}(\omega)\right|^2 \doteq \left|\int_{t\in T} X(t)e^{-i\omega t}\right|^2, \quad \omega \in \Omega,$$

$$\textbf{Recover} \quad X.$$

Can *discretize* this problem as:

$$\textbf{Given} \quad \left|\widehat{X}\right|^2 \in \mathbb{R}^{m\times m},$$

$$\textbf{Recover} \quad X \in \mathbb{R}^{n\times n}.$$

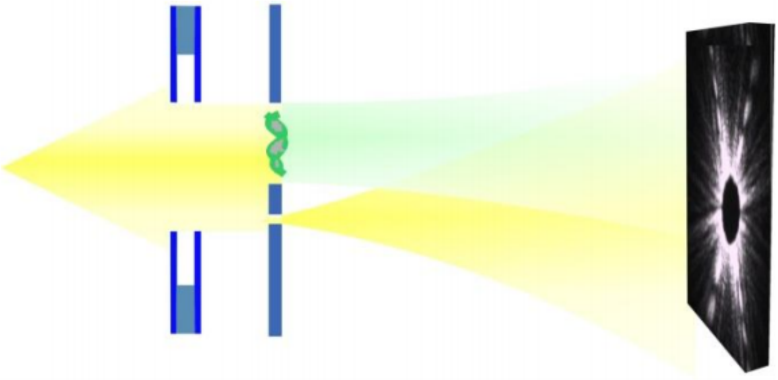
Beamstop

More realistically:

$$\textbf{Given} \quad \left|B \odot \widehat{X}\right|^2 \in \mathbb{R}^{m\times m},$$
$$\textbf{Recover} \quad X \in \mathbb{R}^{n\times n},$$

where B models a *beamstop* which zeros out low-frequencies.

Holographic CDI



The Holographic Phase Retrieval Problem

$$\textbf{Given} \quad R \in \mathbb{R}^{n\times n}, \quad \left| B \odot \left(\widehat{X} + \widehat{R} \right) \right|^2 \in \mathbb{R}^{m\times m},$$

$$\textbf{Recover} \quad X \in \mathbb{R}^{n\times n}.$$

Poisson shot noise model

Quantum mechanics $\rightarrow$ # of photons emitted by an X-ray source is random (Poisson process)

$Y = \left| B \odot \left(\widehat{X} + \widehat{R} \right) \right|^2$

Np: total number of photons reaching detector

$Np = Npp \times$ (number of detector pixels), Npp: photons/pixel

$$\widetilde{Y} \sim_{\text{ind}} \frac{\|Y\|_1}{N_p} \text{Pois}\Big(\frac{N_p}{\|Y\|_1} Y \Big).$$

SNR

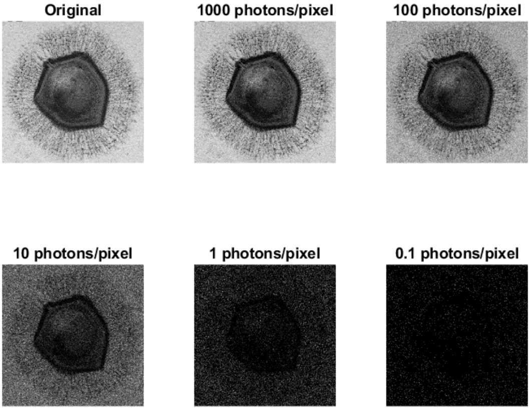
Recall

$$\mathsf{SNR} = \frac{\left\|\mathsf{vec}(Y)\right\|_2^2}{\left\|\mathsf{vec}(\widetilde{Y} - Y)\right\|_2^2}.$$

It follows that

$$\mathbb{E}(\mathsf{SNR}) = Np\frac{\left\|\mathsf{vec}(Y)\right\|_2^2}{\left\|\mathsf{vec}(Y)\right\|_1^2}.$$

Example



SNR values are 1169, 117, 11.7, 1.7, 0.17, respectively.

Poisson Maximum Likelihood Estimation

Given data is $\widetilde{Y}$, where

$$\widetilde{Y} \sim_{\text{ind}} \frac{\|Y\|_1}{N_p} \text{Pois}\Big(\frac{N_p}{\|Y\|_1} Y\Big),$$

for some *unknown* Y.

Poisson Maximum Likelihood Estimation

Given data is $\widetilde{Y}$, where

$$\widetilde{Y} \sim_{\text{ind}} \frac{\|Y\|_1}{N_p} \text{Pois}\Big(\frac{N_p}{\|Y\|_1} Y\Big),$$

for some *unknown* Y.

MLE idea: Find X which gives $Y(X) = \left| B \odot \left(\widehat{X} + \widehat{R}\right) \right|^2$ that makes $\widetilde{Y}$ the most likely data observed.

$\rightarrow$ Combining the MLE and phase retrieval problems into one.

Log-Likelihood Function

$$f(X) = \sum_{i,j=\{1,\ldots,m\}} \Big(Y(X)_{ij} - \widetilde{Y}_{ij} \log(Y(X)_{ij}) \Big) + C.$$

Interlude: TV-Regularization

Add a total variation (TV) regularization term to promote smoothness:

$$\mathrm{TV}(X) \doteq \|\nabla_x X\|_1 + \left\|\nabla_y X\right\|_1.$$

For "discrete derivatives": $\nabla_x X = S_x * X$, $\nabla_y X = S_y * X$, where S_x and S_y are *Sobel filters*.

Regularized Log-Likelihood Function

$$f(X) = \sum_{i,j=\{1,\ldots,m\}} \Big(Y(X)_{ij} - \widetilde{Y}_{ij} \mathsf{log}(Y(X)_{ij}) \Big) + \lambda \mathrm{TV}(X).$$

Regularized Log-Likelihood Function

$$f(X) = \sum_{i,j=\{1,\ldots,m\}} \Big(Y(X)_{ij} - \widetilde{Y}_{ij} \log(Y(X)_{ij}) \Big) + \lambda \mathrm{TV}(X).$$

$\rightarrow$ highly nonconvex function!

Variable Splitting

Can write optimization problem as:

$$\min_{X,U,G_x,G_y} \frac{1}{2} \sum_{(i,j)\in\{1,\dots,m\}^2} \left(\left|U_{ij}\right|^2 - \widetilde{Y}_{ij} \log\left|U_{ij}\right|^2\right) + \lambda \left(\left\|G_x\right\|_1 + \left\|G_y\right\|_1\right)$$

$$\text{subject to} \quad U = B \odot \left(\widehat{X} + \widehat{R}\right),\ G_x = \nabla_x X, G_y = \nabla_y X.$$

Variable splitting with linear constraints

$\rightarrow$ lends itself to ADMM algorithm

ADMM Algorithm

Alternating Direction Method of Multipliers (ADMM)

- Formulate optimization problem using multiple variables subject to linear constraints.
- Update primal variables via single-variable minimization of the augmented Lagrangian.
- Update dual variables to satisfy optimality conditions (for linear constraints).

Applying ADMM

Augmented Lagrangian:

$$\begin{aligned}
&\mathcal{L}\left(X, U, G_x, G_y; V, J_x, J_y\right) \doteq \\
&\frac{1}{2} \sum_{(i,j)\in\{1,\ldots,m\}^2} \left(\left|U_{ij}\right|^2 - \widetilde{Y}_{ij} \log\left|U_{ij}\right|^2\right) + \lambda \left(\left\|G_x\right\|_1 + \left\|G_y\right\|_1\right) \\
&+ \Re\left\langle U - \mathcal{A}\left(X\right) - B, V\right\rangle + \frac{\rho}{2}\left\|U - \mathcal{A}\left(X\right) - B\right\| F^2 \\
&+ \Re\left\langle G_x - \nabla_x X, J_x\right\rangle + \frac{\rho}{2}\left\|G_x - \nabla_x X\right\| F^2 \\
&+ \Re\left\langle G_y - \nabla_y X, J_y\right\rangle + \frac{\rho}{2}\left\|G_y - \nabla_y X\right\| F^2.
\end{aligned}$$

Applying ADMM

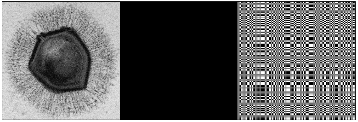
- $X \leftarrow \text{argmin}_X \mathcal{L}(X, U, G_x, G_y; V, J_x, J_y)$
- $U \leftarrow \text{argmin}_U \mathcal{L}\left(X, U, G_x, G_y; V, J_x, J_y\right)$
- $G_x \leftarrow \text{argmin}_{G_x} \mathcal{L}\left(X, U, G_x, G_y; V, J_x, J_y\right)$
- $G_y \leftarrow \text{argmin}_{G_y} \mathcal{L}\left(X, U, G_x, G_y; V, J_x, J_y\right)$
- $V \leftarrow V + \rho\left(U - \mathcal{A}(W) - B\right)$
- $J_x \leftarrow J_x + \rho\left(G_x - \nabla_x W\right)$
- $J_y \leftarrow J_y + \rho\left(G_y - \nabla_y W\right)$.

Test Cases

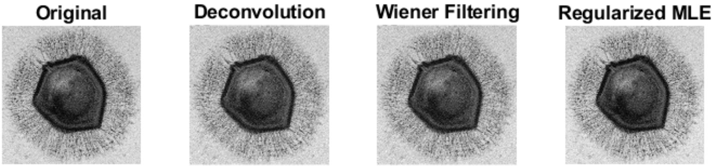
- URA reference, no beamstop
- URA reference, with beamstop
- Block reference, no beamstop
- Block reference, with beamstop

Numerical Simulations - URA Reference, No Beamstop

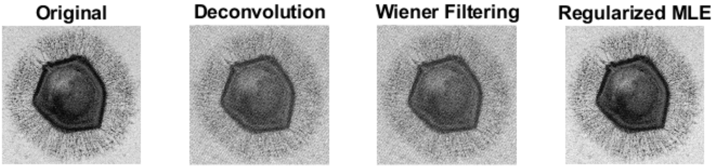
- X : mimivirus, R : URA (uniformly redundant array) reference
- $X, 0, R \in \mathbb{R}^{256\times 256}$
- $Y = [\widehat{X, 0, R}]$, with $2\times$ oversampling
- $$\widetilde{Y} \sim_{\text{ind}} \frac{\|Y\|_1}{N_p} \text{Pois}\Big(\frac{N_p}{\|Y\|_1} Y\Big),$$
$$N_p = (\#\text{ of detector pixels}) \times N_{pp}$$
- Test cases: $N_{pp} = 10^3, 10^2, 10, 1, 0.1$



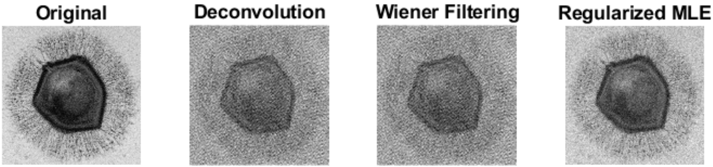
$N_{pp} = 10^3$


Original
Deconvolution
Wiener Filtering
Regularized MLE

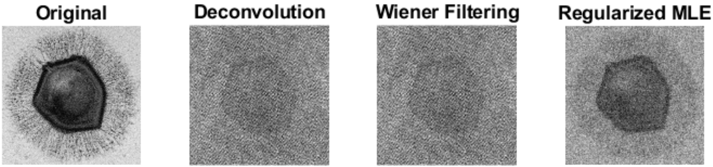
$N_{pp} = 10^2$


Original
Deconvolution
Wiener Filtering
Regularized MLE

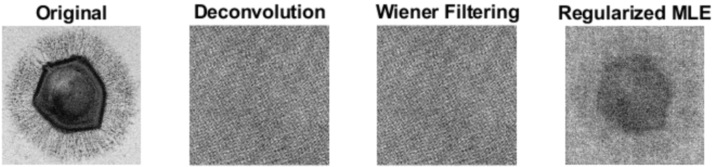
$N_{pp} = 10$


Original
Deconvolution
Wiener Filtering
Regularized MLE

$N_{pp} = 1$

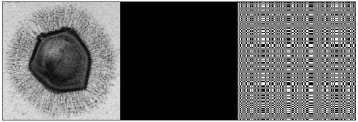

Original
Deconvolution
Wiener Filtering
Regularized MLE

$N_{pp} = 0.1$

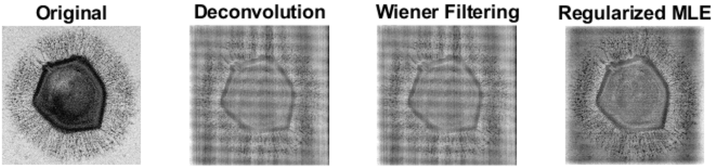

Original
Deconvolution
Wiener Filtering
Regularized MLE

Numerical Simulations - URA Reference, With Beamstop

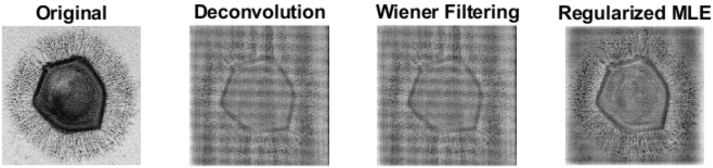
- B : beamstop of size 51×51 (5% area)
- $Y = B \odot \widehat{[X, 0, R]}$, with $2\times$ oversampling
- $$\widetilde{Y} \sim_{\text{ind}} \frac{\|Y\|_1}{N_p} \text{Pois}\Big(\frac{N_p}{\|Y\|_1} Y\Big),$$ $$N_p = (\#\text{ of detector pixels}) \times N_{pp}$$
- Test cases: $N_{pp} = 10^3, 10^2, 10, 1, 0.1$



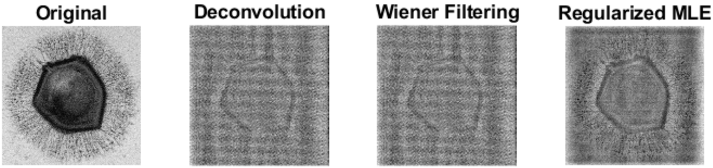
$N_{pp} = 10^3$


Original
Deconvolution
Wiener Filtering
Regularized MLE

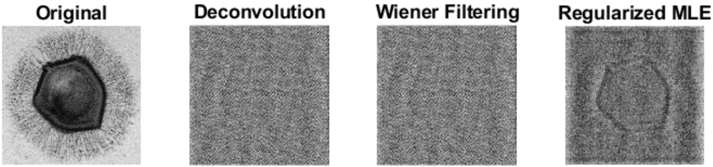
$N_{pp} = 10^2$


Original
Deconvolution
Wiener Filtering
Regularized MLE

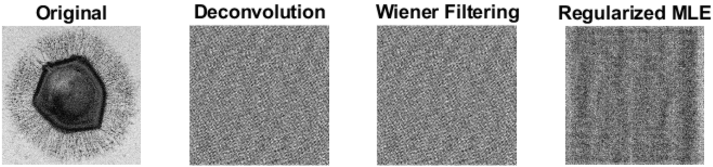
$N_{pp} = 10$


Original
Deconvolution
Wiener Filtering
Regularized MLE

$N_{pp} = 1$


Original
Deconvolution
Wiener Filtering
Regularized MLE

$N_{pp} = 0.1$


Original
Deconvolution
Wiener Filtering
Regularized MLE

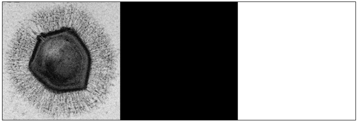
Numerical Simulations - Block Reference, No Beamstop

- X : mimivirus, R : block reference
- $X, 0, R \in \mathbb{R}^{256\times256}$
- $Y = [\widehat{X, 0, R}]$, with $2\times$ oversampling
-

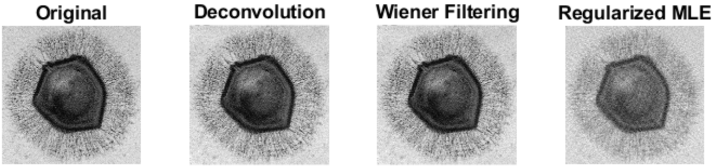
$$\widetilde{Y} \sim_{\text{ind}} \frac{\|Y\|_1}{N_p} \text{Pois}\Big(\frac{N_p}{\|Y\|_1} Y\Big),$$

$$N_p = (\text{\# of detector pixels}) \times N_{pp}$$

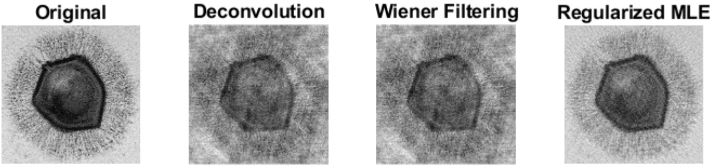
- Test cases: $N_{pp} = 10^6, 10^4, 10^3, 10^2, 10, 1, 0.1$



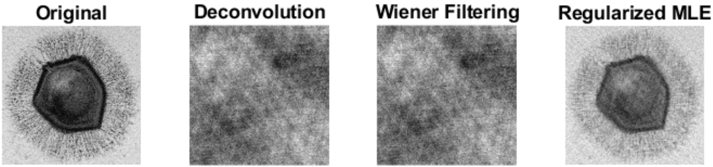
$N_{pp} = 10^6$


Original
Deconvolution
Wiener Filtering
Regularized MLE

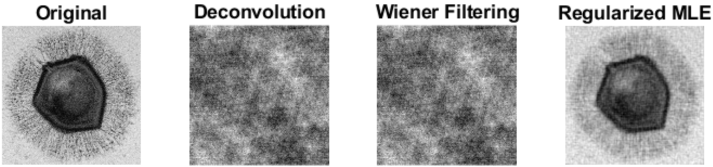
$N_{pp} = 10^4$


Original
Deconvolution
Wiener Filtering
Regularized MLE

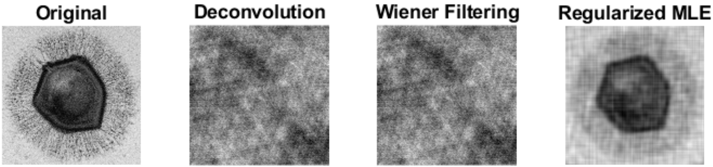
$N_{pp} = 10^2$


Original
Deconvolution
Wiener Filtering
Regularized MLE

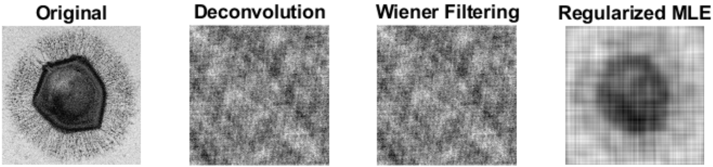
$N_{pp} = 10$


Original
Deconvolution
Wiener Filtering
Regularized MLE

$N_{pp} = 1$

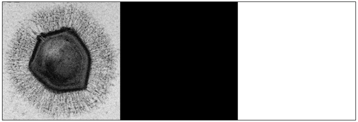

Original
Deconvolution
Wiener Filtering
Regularized MLE

$N_{pp} = 0.1$

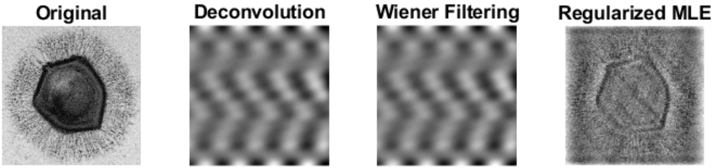

Original
Deconvolution
Wiener Filtering
Regularized MLE

Numerical Simulations - Block Reference, With Beamstop

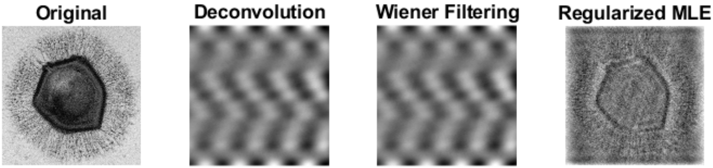
- B : beamstop of size 51×51 (5% area)
- $Y = B \odot \widehat{[X, 0, R]}$, with $2\times$ oversampling
- $$\widetilde{Y} \sim_{\text{ind}} \frac{\|Y\|_1}{N_p} \text{Pois}\Big(\frac{N_p}{\|Y\|_1} Y\Big),$$
  $$N_p = (\#\text{ of detector pixels}) \times N_{pp}$$
- Test cases: $N_{pp} = 10^6, 10^4, 10^3, 10^2, 10, 1, 0.1$



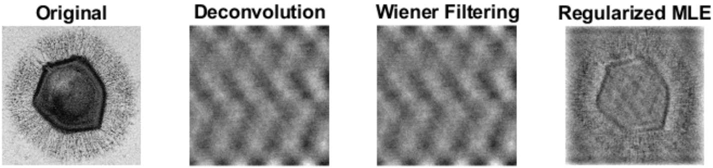
$N_{pp} = 10^6$


Original
Deconvolution
Wiener Filtering
Regularized MLE

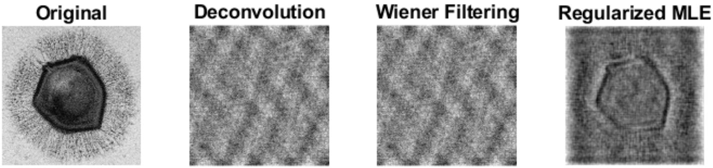
$N_{pp} = 10^4$


Original
Deconvolution
Wiener Filtering
Regularized MLE

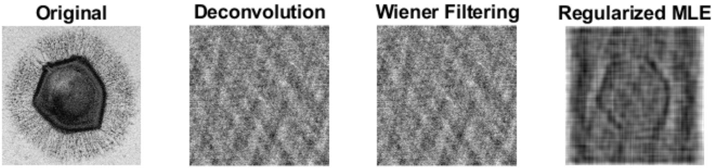
$N_{pp} = 10^2$


Original
Deconvolution
Wiener Filtering
Regularized MLE

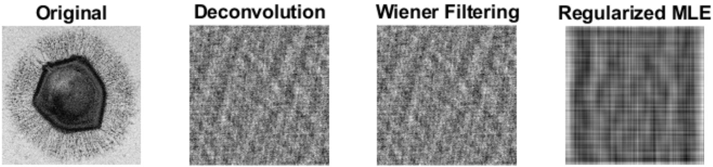
$N_{pp} = 10$


Original
Deconvolution
Wiener Filtering
Regularized MLE

$N_{pp} = 1$


Original
Deconvolution
Wiener Filtering
Regularized MLE

$N_{pp} = 0.1$


Original
Deconvolution
Wiener Filtering
Regularized MLE

Future Work

- Paper in progress
- Experiments with real data

Thank you!